

Querying Metrics from Prometheus

🐧 2 minute read 🛮 🗸 page test

Before you begin

Querying Istio metrics

About the Prometheus addon

Cleanup

See also

This task shows you how to query for Istio Metrics using Prometheus. As part of this task, you will use the web-based interface for querying metric values.

The Bookinfo sample application is used as the example application throughout this task.

Before you begin

- Install Istio in your cluster.
- Install the Prometheus Addon.

Deploy the Bookinfo application.

Querying Istio metrics

Verify that the prometheus service is running in your cluster.
 In Kubernetes environments, execute the following

2. Send traffic to the mesh.

command:

http://\$GATEWAY_URL/productpage in your web browser or issue the following command:

\$\text{curl "http://\$GATEWAY_URL/productpage"}}\$

```
$$\text{$\text{gateway_url is the value set in the Bookinfo}}$$ example.
```

3. Open the Prometheus UI.

For the Bookinfo sample, visit

In Kubernetes environments, execute the following command:

```
$ istioctl dashboard prometheus
```

Click **Graph** to the right of Prometheus in the header.

4. Execute a Prometheus query.

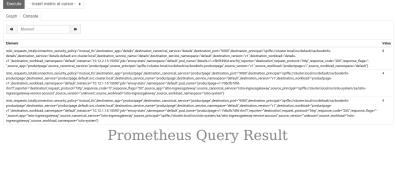
In the "Expression" input box at the top of the web page, enter the text:

```
istio_requests_total
```

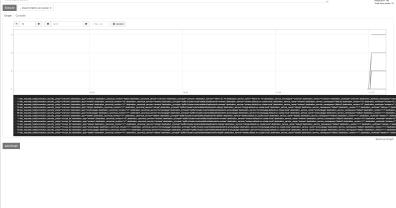
Then, click the **Execute** button.

The results will be similar to:

Prometheus Alerts Graph Status • Help



You can also see the query results graphically by selecting the Graph tab underneath the **Execute** button.



Prometheus Query Result - Graphical

Other queries to try:

• Total count of all requests to the productpage service:

```
istio_requests_total{destination_service="productpage.default.svc.clus
ter.local"}
```

• Total count of all requests to v3 of the reviews service:

```
local", destination_version="v3"}

This query returns the current total count of all requests to
```

istio_requests_total{destination_service="reviews.default.svc.cluster.

This query returns the current total count of all requests to the v3 of the reviews service.

 Rate of requests over the past 5 minutes to all instances of the productpage service: rate(istio_requests_total{destination_service=~"productpage.*", respon se_code="200"}[5m])

About the Prometheus addon

The Prometheus addon is a Prometheus server that comes preconfigured to scrape Istio endpoints to collect metrics. It provides a mechanism for persistent storage and querying of Istio metrics.

For more on querying Prometheus, please read their querying docs.

Cleanup

 Remove any istioctl processes that may still be running using control-C or:

```
$ killall istioctl
```

 If you are not planning to explore any follow-on tasks, refer to the Bookinfo cleanup instructions to shutdown the application.